

## LIST OF CLAIMS / AMENDMENTS

Please amend claims 1, 15, 17, and 23 as shown herein.

Claims 1-27 are pending and are listed following:

1. **(currently amended)** A method for building a data overlay, comprising:

providing a distributed hash table (DHT) that governs the insertion and retrieval of objects into and from a peer-to-peer system, wherein the distributed hash table includes a logical space including a plurality of DHT nodes having an associated plurality of DHT zones; and

building the data overlay as a data structure on top of the logical space of the distributed hash table by associating objects in the data structure with the DHT nodes, and by establishing links between the objects in the data structure, wherein the data structure facilitates dissemination of information to the DHT nodes and gathering of information from the DHT nodes.

2. **(original)** The method according to claim 1, wherein each link includes:

a first field that provides a hardwired pointer that points from a first object to a second object; and

a second field that provides a soft-state pointer that points from the first object to a DHT node which hosts the second object.

1           **3. (original)**   The method according to claim 1, wherein the building  
2 of the data overlay makes use of:

3           a first primitive for setting a reference that establishes a pointer to an object  
4 in the distributed hash table;

5           a second primitive for returning an object referenced by a pointer; and

6           a third primitive for deleting an object referenced by a pointer.  
7

8           **4. (original)**   The method according to claim 1, wherein the data  
9 overlay has a topology of a tree, the tree having a plurality of tree nodes associated  
10 with respective DHT nodes, wherein each tree node has a respective tree node  
11 zone associated therewith which corresponds to a part of the logical space of the  
12 distributed hash table.  
13

14           **5. (original)**   The method according to claim 4, wherein each tree  
15 node in the data overlay includes a key member which identifies a key associated  
16 with its respective tree node zone.  
17

18           **6. (original)**   The method according to claim 5, wherein the key has  
19 a value that is a function of coordinates that identify the center of the respective  
20 tree node zone.  
21

22           **7. (original)**   The method according to claim 4, wherein each tree  
23 node in the data overlay includes an operation member which defines an operation  
24 that is to be performed on data that is passed through the tree node.  
25

1           **8. (original)**   The method according to claim 4, wherein each tree  
2 node in the data overlay includes a report member which defines a report type that  
3 is to be generated using the tree node.

4  
5           **9. (original)**   The method according to claim 4, wherein the building  
6 of the data overlay comprises:

7           establishing a root tree node, the root tree node having a tree node zone  
8 corresponding to an entire span of the logical space of the distributed hash table.

9  
10          **10. (original)**   The method according to claim 4, wherein the building  
11 of the data overlay comprises:

12          examining a tree node zone associated with a particular tree node to  
13 determine whether the tree node zone is smaller than or equal to a DHT zone  
14 associated with the particular tree node's hosting DHT node; and

15          adding a child node associated with the particular tree node if the examining  
16 determines that the tree node zone is not smaller than or equal to the associated  
17 DHT zone.

18  
19          **11. (original)**   The method according to claim 10, further comprising  
20 repeating the examining and the adding for each tree node in the tree.

21  
22          **12. (original)**   A computer readable store including machine readable  
23 instructions for implementing the building of objects in the data overlay according  
24 to the method of claim 10.  
25

1           **13. (original)**   A computer readable store having stored thereon a data  
2 overlay produced according to the method of claim 1.

3  
4           **14. (original)**   A computer readable store having stored thereon a data  
5 overlay having the topology of a tree produced according to the method of claim 4.

6  
7           **15. (currently amended)**   A computer readable store having stored  
8 thereon a data structure, comprising:

9           a logical space of a distributed hash table (DHT), including a plurality of  
10 DHT nodes having a plurality of associated DHT zones, wherein the distributed  
11 hash table governs the insertion and retrieval of objects into and from a peer-to-  
12 peer system;

13           a data overlay implemented as a data structure on top of the logical space of  
14 the distributed hash table logical space, wherein the data overlay uses services  
15 provided by the distributed hash table in routing from one object to another in the  
16 data structure, and wherein the data structure facilitates dissemination of  
17 information to the DHT nodes and gathering of information from the DHT nodes.

18  
19           **16. (original)**   The distributed computer readable store of claim 15,  
20 wherein the data overlay has a topology of a tree, the tree having a plurality of tree  
21 nodes associated with respective DHT nodes, wherein each tree node has a  
22 respective tree node zone associated therewith which corresponds to a part of the  
23 logical space of the distributed hash table.

1  
2       **17. (currently amended)**       A method for passing data through a data  
3 overlay, comprising:

4       providing a distributed hash table (DHT) that governs the insertion and  
5 retrieval of objects into and from a peer-to-peer system, wherein the distributed  
6 hash table includes a logical space including a plurality of DHT nodes having a  
7 plurality of associated DHT zones;

8       building a data overlay as a data structure on top of the logical space of the  
9 distributed hash table by associating objects in the data structure with the DHT  
10 nodes, and by establishing links between the objects in the data structure, wherein  
11 the data overlay defines a plurality of interconnected nodes, and wherein the data  
12 structure facilitates dissemination of information to the DHT nodes and gathering  
13 of information from the DHT nodes; and

14       routing data through the data overlay by passing the data through its  
15 interconnected nodes.

16  
17       **18. (original)**       The method according to claim 17, wherein the data  
18 overlay has a topology of a tree, the tree having a plurality of tree nodes associated  
19 with respective DHT nodes, wherein each tree node has a respective tree node  
20 zone associated therewith which corresponds to a part of the logical space of the  
21 distributed hash table.

22  
23       **19. (original)**       The method according to claim 18, wherein the routing  
24 of data through the data overlay includes gathering data from DHT nodes and  
25 passing the data up through the tree nodes to a root node of the tree.

1  
2       **20. (original)**   The method according to claim 18, wherein the routing  
3 of data through the data overlay includes disseminating data from a root node of  
4 the tree, through the tree nodes, to the DHT nodes.  
5

6       **21. (original)**   The method according to claim 18, wherein each tree  
7 node includes an operation member which defines an operation that is to be  
8 performed on data that is passed through the tree node.  
9

10       **22. (original)**   A computer readable store including machine readable  
11 instructions for implementing the routing of data through the data overlay  
12 according to the method of claim 17.  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25

1           **23. (currently amended)**     A peer-to-peer system including a  
2 plurality of machines interacting in peer-to-peer fashion, comprising:

3           a logical space of a distributed hash table (DHT), including a plurality of  
4 DHT nodes having a plurality of associated DHT zones, wherein the distributed  
5 hash table governs the insertion and retrieval of objects into and from the peer-to-  
6 peer system; and

7           a data overlay implemented as a data structure on top of the logical space  
8 of the distributed hash table, wherein the data overlay uses services provided by  
9 the distributed hash table in routing from one object to another in the data  
10 structure, and wherein the data structure facilitates dissemination of information to  
11 the DHT nodes and gathering of information from the DHT nodes,

12           wherein the logical space of the distributed hash table and the data overlay  
13 are implemented in distributed fashion in respective stores of the plurality of  
14 machines in the peer-to-peer system.

15  
16           **24. (original)**     The system according to claim 23, wherein the data  
17 overlay has a topology of a tree, the tree having a plurality of tree nodes associated  
18 with respective DHT nodes, wherein each tree node has a respective tree node  
19 zone associated therewith which corresponds to a part of the logical space of the  
20 distributed hash table.

21  
22           **25. (original)**     The system according to claim 24, further including  
23 routing logic configured to route data through the data overlay by passing the data  
24 through the tree nodes.

25

1           **26. (original)**   The system according to claim 25, wherein the routing  
2 logic is configured to route the data through the data overlay by gathering data  
3 from DHT nodes and passing the data up through the tree nodes to a root node of  
4 the tree.

5  
6           **27. (original)**   The system according to claim 25, wherein the routing  
7 logic is configured to route data through the data overlay by disseminating data  
8 from a root node of the tree, through the tree nodes, to the DHT nodes.